# Scientist's Challenge 20161109 - Collaborating on Software Projects

The Scientist's Challenge for November comes from Jeremiah Lant, Hydrologist at the Kentucky Water Science Center.

Some questions about **collaborating on software projects** for CDI to ponder, discuss, and share at a future Monthly Meeting:

- What collaborative development models and workflows are scientists using for software development?
    - Fork and pull model
    - Shared repository model
    - Other models?
- How are these models and workflows implemented in practice?
- How are scientists informing/teaching other fellow scientists about what works?

Here are the links to the publication referenced in the slides: ***How open science helps researchers succeed***, McKiernan et al., 2016 (html) (pdf)

*There are many ways you can contribute to this challenge: suggest ideas or recommendations directly as a comment to this forum (you must be logged in), email Jeremiah directly at jlant@usgs.gov, or email cdi@usgs.gov and the CDI coordinators will pass on the information.*

2016-11-09-cdi-scientist-challenge-jeremiah-lant.pdf

## The Challenge

- Learning proper software collaboration models and workflows.
- Implementing proper software collaboration models and workflows.

## Software Tool Suite



GitLab

Atlassian
Bitbucket

## Collaborative Development Models

***Fork and Pull***

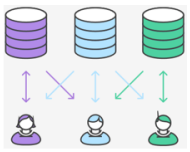
- Collaborators *fork* an existing repository and work on his/her own fork of the source repository.
- Collaborators do not need access to the source repository.
- Collaborators contribute to the source repository via *pull requests*.
- The project maintainer of the source repository reviews pull requests from collaborators and can *pull* (merge) the changes into the source repository.
- Popular with open source projects because it reduces the amount of friction for new contributors and allows people to work independently without upfront coordination.
- Distributed

***Shared repository***

- Collaborators are granted access to *push* to a single shared repository.
- Collaborators have common access to a blessed repository where all the developers can push too.
- Collaborators must agree on the branch and merge convention.
- *Pull requests* initiate code review and general discussion about a set of changes before the changes are merged into the main development branch.
- Popular with small teams and organizations collaborating on private projects.
- Centralized

## Collaborative Development Models

***Fork and Pull***



***Shared repository***

## Collaborative Development Models

***Fork and Pull***

***Shared repository***

Gitflow Workflow

## Gitflow Workflow

## Questions?

- What collaborative development models and workflows are scientist's using for software development?
  - *Fork and pull model*
  - *Shared repository model*
  - Other models?
- How are these models and workflows implemented in practice?
- How are scientist's informing/teaching other fellow scientist's about what works?